DEPARTMENT OF DEFENCE

DEFENCE SCIENCE & TECHNOLOGY ORGANISATION

DSTO

Coordinate Usage in the Virtual Ship Architecture Issue 1.00

John P. Best

DSTO-GD-0260

Coordinate Usage in the Virtual Ship Architecture
Issue 1.00

John P. Best

Maritime Operations Division
Aeronautical and Maritime Research Laboratory

DSTO-GD-0260

ABSTRACT

This document describes the coordinate systems used within the Virtual Ship Architecture (VSA) and defines object attributes that convey kinematic information.

RELEASE LIMITATION

Approved for public release

DEPARTMENT OF DEFENCE
DEFENCE SCIENCE & TECHNOLOGY ORGANISATION
DSTO

AQ F01-07-1351

Published by

DSTO Aeronautical and Maritime Research Laboratory
PO Box 4331
Melbourne Victoria 3001 Australia

Telephone: (03) 9626 7000
Fax: (03) 9626 7999

AR-011-626
October 2000

APPROVED FOR PUBLIC RELEASE

Coordinate Usage in the Virtual Ship Architecture Issue 1.00

Executive Summary

The Virtual Ship Architecture (VSA) provides a framework for conducting distributed simulation in the maritime domain. It is based upon the High Level Architecture (HLA) and supports the linking of simulations that represent complete entities, such as warships, submarines and aircraft, or simulations that represent the systems of which these entities are composed. These include sensor, weapon, countermeasure, navigation and command and control systems. It is through linking these simulations that a virtual representation of a warship may be created.

An essential requirement for distributed simulation components to be linked and operate together is to exchange information about the kinematic attributes of the entities that are modelled. The kinematic attributes include position, velocity, acceleration and those attributes that specify orientation. These attributes must be provided in a consistent way, with respect to a common frame of reference.

A key part of the Virtual Ship Architecture is therefore the adoption of a common frame of reference with respect to which the kinematic attributes of entities are transmitted amongst the participating simulations.

The common frame of reference used in the Virtual Ship Architecture is the WGS84 coordinate system. All entities in the simulation are considered to have a coordinate system fixed to them. The origin of this coordinate system specifies the location of an entity. The orientation of this coordinate system with respect to the reference specifies the entity's orientation.

Page deliberately left blank

Revision Record

The controlled version of this document resides in the Virtual Ship Project System Analysis and Design Folder. Any copy of this document, electronic or otherwise, should be considered uncontrolled. Amendment shall be by whole document replacement. Proposals for change should be forwarded to the issuing authority.

Issue	Date	Author	Description of Revision
1.00	20Jul00	J. Best	First issue.

Page deliberately left blank

Contents

Page deliberately left blank

1. Introduction

The Virtual Ship Architecture (VSA) provides a framework for conducting distributed simulation in the maritime domain. It is based upon the High Level Architecture (HLA) and supports the linking of simulations that represent complete entities, such as warships, submarines and aircraft, or simulations that represent the systems of which these entities are composed. These include sensor, weapon, countermeasure, navigation and command and control systems. It is through linking these simulations that a virtual representation of a warship may be created. The Virtual Ship Architecture Description Document (VSADD) [1] provides a comprehensive description of the VSA and should be read in conjunction with this document.

An essential requirement for distributed simulation components to be linked and operate together is to exchange information about the kinematic attributes of the entities that are modelled. The kinematic attributes include position, velocity, acceleration and those attributes that specify orientation. These attributes must be provided in a consistent way, with respect to a common frame of reference. Upon receipt of attribute updates, each federate will typically need to transform these data to coordinate systems that are local to the entities it represents.

A key part of the Virtual Ship Architecture is therefore the adoption of a common frame of reference with respect to which the kinematic attributes of entities are transmitted amongst the participating federates. This document describes the common frame of reference used within the Virtual Ship Architecture as a basis for providing kinematic attributes of entities.

2. The reference coordinate systems

The fundamental reference coordinate system is a right-handed Cartesian coordinate system whose origin is located at the centre of mass of the earth. The coordinate system is considered fixed within the earth, so it rotates with it. In this coordinate system the z-axis is defined by the Conventional Internation Origin (CIO) which is the mean position of the earth's rotation axis in the period 1900-1905 [2]. The x-axis is defined as passing through the mean Greenwich meridian and the y-axis completes a right-handed coordinate system. This is illustrated in Figure 1.

This coordinate system is used extensively in geodesy. A principal use is to define ellipsoids that approximate the shape of the earth. Typically these are oblate spheriods given by the expression

$$\frac{x^2}{a^2} + \frac{y^2}{a^2} + \frac{z^2}{b^2} = 1. \tag{1}$$

In this expression a is the semi-major axis and b is the semi-minor axis. Other quantities used in reference to these ellipsoids are the flattening

$$f = (a - b)/a, \tag{2}$$

and the eccentricity

$$e^2 = 1 - b^2/a^2 = 2f - f^2. \tag{3}$$

A commonly used ellipsoid in the military context is that which define the WGS84 coor-

dinate system [3]. The parameters that define this ellipsoid are

$$
\begin{aligned}
a &= 6378137.00000\,\mathrm{m},\\
b &= 6356752.31425\,\mathrm{m},\\
1/f &= 298.2572235614,\\
e^2 &= 0.00669437999013.
\end{aligned}
$$

In the WGS84 coordinate system the position of a point is typically given as a latitude ϕ, longitude λ and height h above the ellipsoid. The meridian plane is defined as the plane containing both the z-axis and the point concerned. The normal to the ellipsoid passing through this point lies in the meridian plane. The latitude is the angle between the normal to the ellipsoid and the $x-y$ plane, measured in the meridian plane. The longitude is the angle between the Greenwich meridian and the meridian plane of the point. The height coordinate is the height of the point above, or below, the ellipsoid, measured along the normal. Typically the latitude is given as $\phi \in [-\pi/2, \pi/2]$ and the longitude is given as $\lambda \in (-\pi, \pi]$. Latitude is measured positive to the North of the equator and negative to the South. The longitude is measured negative to the West and positive to the East of the Greenwich meridian.

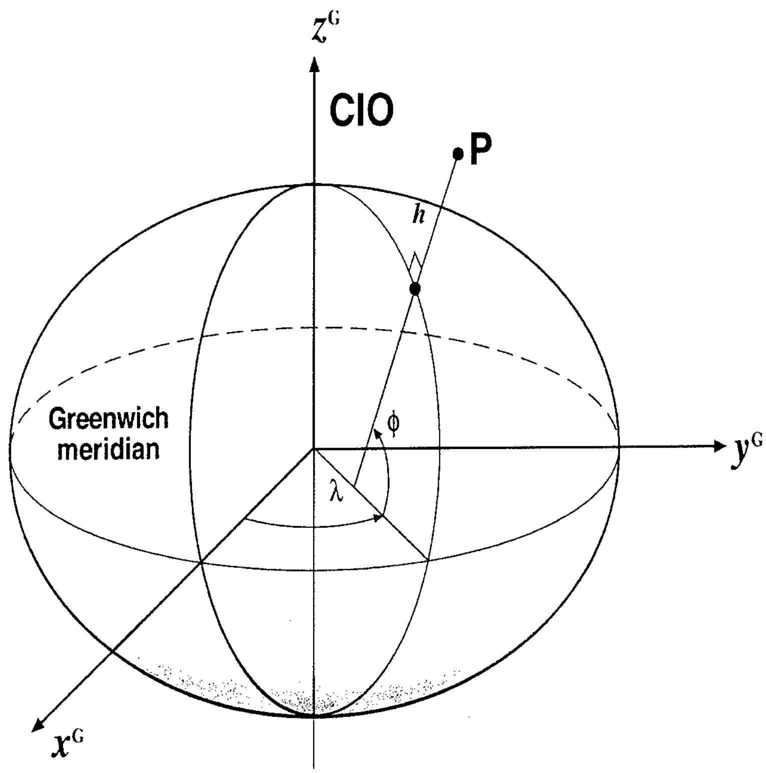

Figure 1: The G coordinate system.

In the remainder of this document this coordinate system is central. We denote it as the G coordinate system. In addition, we will use superscripts to denote quantities that are given with respect to this coordinate system. For example, the position vector of a point given with respect to this coordinate system will be donted as $\boldsymbol{r}^G$.

For a point positioned on, or near, the Earth's surface it is often appropriate to work with a local coordinate system. Let us define a coordinate system about some point. We will call this the local geodetic coordinate system, and denote it as LG. It is illustrated in Figure 2. The z^{LG}-axis is defined by the exterior normal to the ellipsoid that passes through the point, the y^{LG}-axis is perpendicular to this and defines, with the z^{LG}-axis, a plane that contains the z^G axis. The y^{LG}-axis is said to define geodetic North. The x^{LG}-axis is chosen to complete a right handed coordinate system and define East. The transformation between the G and LG coordinate system is given in Appendix B. Note that the LG coordinate system may be defined at any point. It need not have its origin on the surface of the ellipsoid.

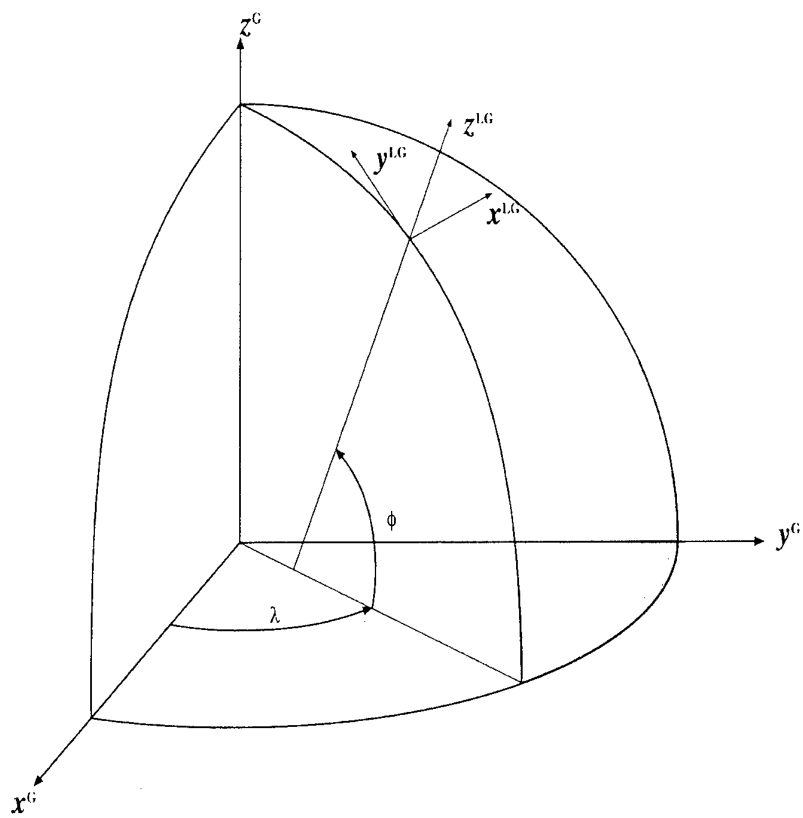

Figure 2: The LG coordinate system.

3. Specifying the kinematic attributes of an entity

Within the Virtual Ship Architecture are notions of entities that occupy a finite volume of space and these include the CompositeEntity, ComponentEntity and Sensor Task. An instance of a CompositeEntity object represents a complete entity within the VSA. Ships, submarines, aircraft, missiles and countermeasures are all examples of CompositeEntity objects. An instance of a ComponentEntity object represents entities that are components of a CompositeEntity. Sensor, weapon, countermeasure, navigation and command and control systems are all examples of ComponentEntity objects. An instance of the SensorTask object represents a particular task being performed by a sensor, such as a radar, ESM or sonar and is characterised, in part, by the sensor beam.

To specify the kinematics of each of these types of entity, it is assumed that they have associated with them a right-handed Cartesian coordinate system. The x-axis of this coordinate system is typically in the direction that would commonly be identified as forwards, sometimes known as the figure axis. The z-axis is in the direction typically identified as up and the y-axis completes a right-handed coordinate system. The origin of the coordinate systems associated with CompositeEntity and ComponentEntity objects is located at the centre of mass of the entity.

For notational purposes, the coordinate system associated with a CompositeEntity object instance shall be denoted by E, that associated with a ComponentEntity object instance shall be denoted by e and that associated with a sensor task shall be denoted by S. These coordinate systems are illustrated in Figures 3 and 4.

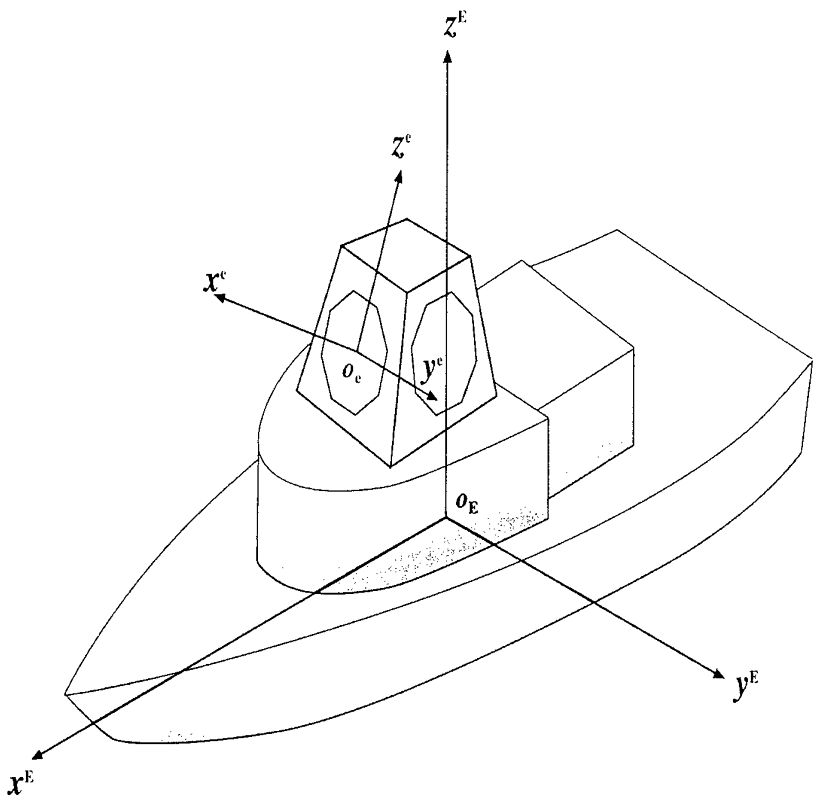

Figure 3: The E and e coordinate systems.

To specify the kinematics of these entities requires that the motion of the origin of the coordinate system be given, and the orientation of the coordinate system given with respect to some reference frame. The motion of the origin is given as a position, velocity and acceleration vector. The notations $o_{C_1}^{C_2}$, $\dot{o}_{C_1}^{C_2}$ and $\ddot{o}_{C_1}^{C_2}$ are used to specify these values. The subscript $_{C_1}$ denotes the coordinate system of which this point is the origin and the superscript ^{C_2} denotes the coordinate system within which the origin is specified. The orientation is specified through the use of Euler angles that define the rotation transformation from the reference coordinate system C_2 to the entity coordinate system C_1. The convention for defining the Euler angles is described in Appendix C and the set of three Euler angles is denoted by $\Omega^{C_2 \rightarrow C_1}$. In addition to specifying the Euler angles their time derivatives may also be given to specify the rotation rate of the entity, and

these are denoted as $\dot{\Omega}^{C_2 \to C_1}$. The relationship between time derivatives of the Euler angles and the angular velocity of a rigid body is given in Appendix F.

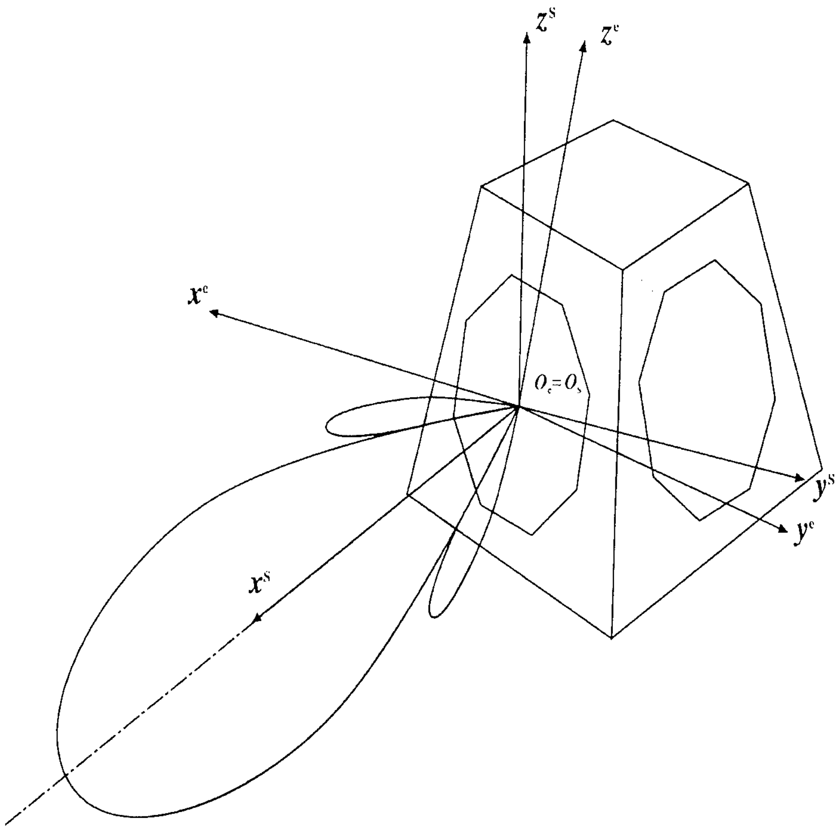

Figure 4: The e and S coordinate systems.

4. Object attributes that convey coordinate data

Within the VS-FOM there are a number of object attributes that convey kinematic information. The remainder of this section describes these attributes.

4.1 The CompositeEntity object class

The composite entity has, among others, the following attributes:

Position,
Velocity,
Acceleration,
Orientation,
OrientationRate.

The Position attribute is the position vector of the origin of the E coordinate system, given with respect to the G system. This is written as

$$\text{Position} = o_E^G.$$

The Velocity attribute is the velocity of the origin of the E coordinate system, given with respect to the G system. This is written as

$$\text{Velocity} = \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}.$$

The Acceleration attribute is the acceleration of the origin of the E coordinate system, given with respect to the G system. This is written as

$$\text{Acceleration} = \ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}.$$

The Orientation attribute is the Euler angle set that defines the rotation transformation from the G system to the E system. This is written as

$$\text{Orientation} = \Omega^{\mathrm{G}\to\mathrm{E}}.$$

The OrientationRate attribute is the time derivative of the Euler angle set that defines the rotation transformation from the G system to the E system. This is written as

$$\text{OrientationRate} = \dot{\Omega}^{\mathrm{G}\to\mathrm{E}}.$$

4.2 The ComponentEntity object class

The component entity has, among others, the following attributes:

RelativePosition,
RelativeOrientation.

The RelativePosition attribute is the position vector of the origin of the e coordinate system, given with respect to the E system of the CompositeEntity object of which this ComponentEntity is a part, otherwise known as the parent entity. This is written as

$$\text{RelativePosition} = \boldsymbol{o}_{\mathrm{e}}^{\mathrm{E}}.$$

The RelativeOrientation attribute is the Euler angle set that defines the rotation transformation from the E system to the e system. This is written as

$$\text{RelativeOrientation} = \Omega^{\mathrm{E}\to\mathrm{e}}.$$

4.3 Sensor tasks

The SensorTask object instance has, among others, the following attributes:

BeamPatternReference,
BeamPatternOrientation,
BeamPatternOrientationRate.

The origin of the beam pattern coordinate system S is assumed to be colocated with the origin of the coordinate system attached to the ComponentEntity with which the SensorTask is associated. The BeamPatternReference attribute specifies the coordinate system with respect to which the orientation of the S coordinate system is given. The BeamPatternOrientation attribute is the Euler angle set that defines the rotation transformation from the reference system to the S system. The BeamPatternOrientationRate attribute provides the time derivative of these Euler angles.

If BeamPatternReference = 1 then the reference system is the G system, so

$$\text{BeamPatternOrientation} = \Omega^{G \to S},$$

and

$$\text{BeamPatternOrientationRate} = \dot{\Omega}^{G \to S}.$$

If BeamPatternReference = 2 then the reference system is the LG system, so

$$\text{BeamPatternOrientation} = \Omega^{LG \to S},$$

and

$$\text{BeamPatternOrientationRate} = \dot{\Omega}^{LG \to S}.$$

If BeamPatternReference = 3 then the reference system is the E system, that is the coordinate system of the CompositeEntity object instance that is the parent of the ComponentEntity with which the task is associated. Hence

$$\text{BeamPatternOrientation} = \Omega^{E \to S},$$

and

$$\text{BeamPatternOrientationRate} = \dot{\Omega}^{E \to S}.$$

The LG system will be appropriately used as the reference in the case of a stabilised sensor and the E system might be used otherwise.

4.4 Tracks

The Track class and its subclasses have various attributes that convey kinematic information. A track is considered to represent a point in space and does not, therefore, require specification of orientation. In the first instance, consider the attributes of the AbsoluteTrack subclass. Those that convey kinematic information are

Position,
PositionError,
Velocity,
VelocityError.

The Position attribute is the position vector of the track given with respect to the G system. The Velocity attribute is the velocity of the track given with respect to the G system. The error attributes are measured with respect to the same coordinate system as the quantities to which they refer.

The attributes of the RelativeTrack subclass that convey kinematic information are

FootpointPosition,
FootpointPositionError,
Bearing,
BearingError,
Elevation,
ElevationError,
Range,
RangeError,
RangeRate,
RangeRateError.

The FootointPosition attribute is the position vector of the point with respect to which the track is specified, given with respect to the G system.

The Bearing attribute is the bearing to the track, given with respect to the LG coordinate system defined at the footpoint. The bearing is the angle between the x^{LG}-axis and the vertical plane that contains the z^{LG}-axis and the track position. This is illustrated in Figure 5.

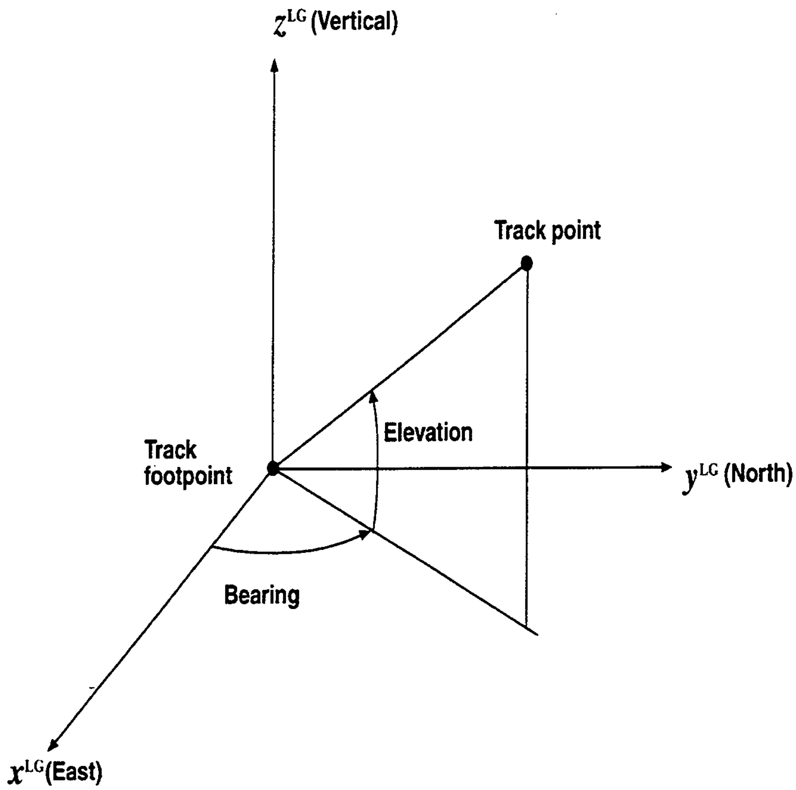

Figure 5: The bearing and elevation attributes of the RelativeTrack object class.

The Elevation attribute is the elevation to the track point, given with respect to the LG coordinate system defined at the footpoint. The elevation is the angle between a line joining the footpoint to the track and the $x^{\mathrm{LG}} - y^{\mathrm{LG}}$ plane.

The Range attribute is the straight-line distance between the footpoint and the track. The RangeRate attribute is the time derivative of this quantity.

The error attributes are measured with respect to the same coordinate system as the quantities to which they refer.

5. Dead-reckoning

To reduce the frequency of certain kinematic attribute updates the technique of dead-reckoning is used. Dead-reckoning involves extrapolating the kinematic attributes of an

entity in time. A federate responsible for updating the kinematic attributes of an entity will typically also dead-reckon them and provide attribute updates when the deviation between the actual and dead-reckoned values exceeds some desired accuracy.

To elaborate by way of an example, suppose a federate provides the following at time t_1:

$$\boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t_1), \quad \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1), \quad \ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1),$$

that is the position, velocity and acceleration of a composite entity. Extrapolating, the dead-reckoned position at time t is

$$_{\mathrm{DR}}\boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t) = \boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t_1) + \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1)(t - t_1) + \tfrac{1}{2}\ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1)(t - t_1)^2. \tag{4}$$

If the position is required to be known throughout the federation with accuracy ϵ, then an attribute update is required only when

$$|\boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t) -_{\mathrm{DR}} \boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t)| > \epsilon. \tag{5}$$

The significant advantage of this approach is that if the acceleration of an entity varies slowly then a large time will elapse before (5) is satisfied and an attribute update is required.

There are a number of approaches to dead-reckoning. These primarily concern which of the attributes should be extrapolated in time and whether they should be extrapolated to first or second order in time. The CompositeEntity attribute DRAlgorithm provides an enumeration of possible algorithms. For each object instance, the federate updating the kinematic attributes of the object will determine which algorithm is appropriate. Federates that subscribe to these attributes will require the capability to utilise any of the dead-reckoning algorithms.

The enumerators One, Two, Three, Four, Five and Six specify which dead-reckoning algorithm to use and these are detailed in the following. It is assumed that the most recent attribute update has occurred at time t_1.

DRAlgorithm = One

$$\begin{aligned}
\boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t) &= \boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t_1), \\
\dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1) \\
\ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1), \\
\Omega^{\mathrm{G}\to\mathrm{E}}(t) &= \Omega^{\mathrm{G}\to\mathrm{E}}(t_1), \\
\dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t) &= \dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t_1).
\end{aligned} \tag{6}$$

DRAlgorithm = Two

$$\begin{aligned}
\boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t) &= \boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t_1) + \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1)(t - t_1), \\
\dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1), \\
\ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1), \\
\Omega^{\mathrm{G}\to\mathrm{E}}(t) &= \Omega^{\mathrm{G}\to\mathrm{E}}(t_1), \\
\dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t) &= \dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t_1).
\end{aligned} \tag{7}$$

DRAlgorithm = Three

$$
\begin{aligned}
\boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t) &= \boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t_1) + \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1)(t - t_1) + \tfrac{1}{2}\ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1)(t - t_1)^2, \\
\dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1) + \ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1)(t - t_1), \\
\ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1), \\
\Omega^{\mathrm{G}\to\mathrm{E}}(t) &= \Omega^{\mathrm{G}\to\mathrm{E}}(t_1), \\
\dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t) &= \dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t_1).
\end{aligned}
\tag{8}
$$

DRAlgorithm = Four

$$
\begin{aligned}
\boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t) &= \boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t_1), \\
\dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1), \\
\ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1), \\
\Omega^{\mathrm{G}\to\mathrm{E}}(t) &= \Omega^{\mathrm{G}\to\mathrm{E}}(t_1) + \dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t_1)(t - t_1), \\
\dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t) &= \dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t_1).
\end{aligned}
\tag{9}
$$

DRAlgorithm = Five

$$
\begin{aligned}
\boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t) &= \boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t_1) + \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1)(t - t_1), \\
\dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1), \\
\ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1), \\
\Omega^{\mathrm{G}\to\mathrm{E}}(t) &= \Omega^{\mathrm{G}\to\mathrm{E}}(t_1) + \dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t_1)(t - t_1), \\
\dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t) &= \dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t_1).
\end{aligned}
\tag{10}
$$

DRAlgorithm = Six

$$
\begin{aligned}
\boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t) &= \boldsymbol{o}_{\mathrm{E}}^{\mathrm{G}}(t_1) + \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1)(t - t_1) + \tfrac{1}{2}\ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1)(t - t_1)^2, \\
\dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \dot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1) + \ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1)(t - t_1), \\
\ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t) &= \ddot{\boldsymbol{o}}_{\mathrm{E}}^{\mathrm{G}}(t_1), \\
\Omega^{\mathrm{G}\to\mathrm{E}}(t) &= \Omega^{\mathrm{G}\to\mathrm{E}}(t_1) + \dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t_1)(t - t_1), \\
\dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t) &= \dot{\Omega}^{\mathrm{G}\to\mathrm{E}}(t_1).
\end{aligned}
\tag{11}
$$

Dead-reckoning is also used to extrapolate the orientation of the coordinate system associated with the beam pattern that defines the SensorTask object. The attribute BeamPatternDRAlgorithm provides an enumerated description of the algorithm used. In the following it is assumed that the coordinate system S is given with respect to the G system. The formulae are the same in the event that the LG or E systems are used as reference.

BeamPatternDRAlgorithm = One

$$
\begin{aligned}
\Omega^{\mathrm{G}\to\mathrm{S}}(t) &= \Omega^{\mathrm{G}\to\mathrm{S}}(t_1), \\
\dot{\Omega}^{\mathrm{G}\to\mathrm{S}}(t) &= \dot{\Omega}^{\mathrm{G}\to\mathrm{S}}(t_1).
\end{aligned}
\tag{12}
$$

BeamPatternDRAlgorithm = Two

$$
\begin{aligned}
\Omega^{\mathrm{G}\to\mathrm{S}}(t) &= \Omega^{\mathrm{G}\to\mathrm{S}}(t_1) + \dot{\Omega}^{\mathrm{G}\to\mathrm{S}}(t_1)(t - t_1), \\
\dot{\Omega}^{\mathrm{G}\to\mathrm{S}}(t) &= \dot{\Omega}^{\mathrm{G}\to\mathrm{S}}(t_1).
\end{aligned}
\tag{13}
$$

6. References

[1] *Virtual Ship Architecture Description Document. Version 1.00.* DSTO-GD-0257. (2000)

[2] Vaniček, P. & Krakiwsky, E. J. *Geodesy: The Concepts.* North-Holland. (1986)

[3] Hofmann-Wellenhof, B. Lichtenegger, H. & Collins, J. *Global Positioning System. Theory and Practice.* Third, revised edition. Springer-Verlag. (1994)

[4] Goldstein, H. *Classical Mechanics.* Addison-Wesley. (1980)

Page deliberately left blank

Appendix A - Transforming between Cartesian coordinates and latitude, longitude and height

The transformation from a representation of a point as a latitude, longitude and height (ϕ, λ, h) to Cartesian coordinates (x, y, z) in the G system is straight forward. We have

$$x = (N + h)\cos\phi\cos\lambda, \tag{A1}$$

$$y = (N + h)\cos\phi\sin\lambda, \tag{A2}$$

$$z = ((1 - e^2)N + h)\sin\phi, \tag{A3}$$

where

$$N = a/\sqrt{1 - e^2\sin^2\phi}. \tag{A4}$$

The inverse transformation is a little more complex. From (A1) and (A2) define

$$p = \sqrt{(x^2 + y^2)} = (N + h)\cos\phi, \tag{A5}$$

noting that $-\pi/2 \leq \phi \leq \phi/2$. Using this in (A3) yields

$$z = (p/\cos\phi - e^2N)\sin\phi. \tag{A6}$$

If we define

$$f(\phi) = p\tan\phi - e^2N\sin\phi - z, \tag{A7}$$

then ϕ is a root of f and may be found using appropriate root finding techniques.

The Newton-Raphson method is a technique that works well in this case. To begin the iterations using this method requires an initial estimate of the root which can be obtained as follows. Using (A5) in (A3) yields

$$\tan\phi\left(1 - \frac{e^2N}{N + h}\right) = z/p. \tag{A8}$$

Now N is of the order of the radius of the Earth, so for objects for which $h \ll N$, $N/(N + h) \approx 1$ and (A8) gives a first approximation to the latitude as

$$\phi^{(0)} = \arctan\left(\frac{z}{p(1 - e^2)}\right). \tag{A9}$$

It should be noted that this approximation is of high accuracy and only one or two interations of the Newton-Raphson method are typically required.

Having determined ϕ it remains to determine λ and h. A convenient way to get λ is to first note that

$$\tan\frac{\lambda}{2} = \frac{\sin(\lambda/2)}{\cos(\lambda/2)} = \frac{2\sin(\lambda/2)\cos(\lambda/2)}{2\cos^2(\lambda/2)} = \frac{\sin\lambda}{\cos\lambda + 1}. \tag{A10}$$

Using (A1), (A2) and the definition of p yields

$$\tan\frac{\lambda}{2} = \frac{y}{x + p}, \tag{A11}$$

so

$$\lambda = 2\arctan\left(\frac{y}{x + p}\right). \tag{A12}$$

h is obtained directly from (A5) as

$$h = p/\cos\phi - N. \tag{A13}$$

Appendix B - Transforming between the G and LG coordinate systems

The unit vectors that define the LG coordinate system are given in terms of the latitude and longitude of the origin of this system. Let this point have Cartesian coordinates $(x^{\mathrm{G}}, y^{\mathrm{G}}, z^{\mathrm{G}})$ with respect to the G coordinate system, which may be alternatively given as (ϕ, λ, h). With reference to Figure 2, we have

$$\begin{aligned} \boldsymbol{i}^{\mathrm{LG}} &= -\sin\lambda\, \boldsymbol{i}^{\mathrm{G}} + \cos\lambda\, \boldsymbol{j}^{\mathrm{G}}, \\ \boldsymbol{j}^{\mathrm{LG}} &= -\sin\phi\cos\lambda\, \boldsymbol{i}^{\mathrm{G}} - \sin\phi\sin\lambda\, \boldsymbol{j}^{\mathrm{G}} + \cos\phi\, \boldsymbol{k}^{\mathrm{G}}, \\ \boldsymbol{k}^{\mathrm{LG}} &= \cos\phi\cos\lambda\, \boldsymbol{i}^{\mathrm{G}} + \cos\phi\sin\lambda\, \boldsymbol{j}^{\mathrm{G}} + \sin\phi\, \boldsymbol{k}^{\mathrm{G}}. \end{aligned} \tag{B1}$$

If $\boldsymbol{e}^{\mathrm{LG}}$ denotes a unit vector with components given with respect to the LG coordinate system, and $\boldsymbol{e}^{\mathrm{G}}$ denotes the same vector with components given with respect to the G system then

$$\boldsymbol{e}^{\mathrm{G}} = R_{\mathrm{LG}\to\mathrm{G}} \boldsymbol{e}^{\mathrm{LG}}, \tag{B2}$$

where the rotation matrix $R_{\mathrm{LG}\to\mathrm{G}}$ is given as

$$R_{\mathrm{LG}\to\mathrm{G}} = \begin{pmatrix} -\sin\lambda & -\sin\phi\cos\lambda & \cos\phi\cos\lambda \\ \cos\lambda & -\sin\phi\sin\lambda & \cos\phi\sin\lambda \\ 0 & \cos\phi & \sin\phi \end{pmatrix}. \tag{B3}$$

From (B2) we also have

$$\boldsymbol{e}^{\mathrm{LG}} = R^{-1}_{\mathrm{LG}\to\mathrm{G}} \boldsymbol{e}^{\mathrm{G}} = R^{T}_{\mathrm{LG}\to\mathrm{G}} \boldsymbol{e}^{\mathrm{G}} = R_{\mathrm{G}\to\mathrm{LG}} \boldsymbol{e}^{\mathrm{G}}, \tag{B4}$$

where the property of rotation matrices that their inverse is equal to the transpose gives us

$$R_{\mathrm{G}\to\mathrm{LG}} = R^{T}_{\mathrm{LG}\to\mathrm{G}}, \tag{B5}$$

or writing it out in full

$$R_{\mathrm{G}\to\mathrm{LG}} = \begin{pmatrix} -\sin\lambda & \cos\lambda & 0 \\ -\sin\phi\cos\lambda & -\sin\phi\sin\lambda & \cos\phi \\ \cos\phi\cos\lambda & \cos\phi\sin\lambda & \sin\phi \end{pmatrix}. \tag{B6}$$

Consider now the transformation of position vectors give with respect to the G and LG coordinate systems. Denote by $\boldsymbol{o}^{\mathrm{G}}_{\mathrm{LG}}$ the position vector of the origin of the LG system, given with respect to the G coordinate system. It is assumed that the origin of the G system is the null vector. Let $\boldsymbol{r}^{\mathrm{G}}$ denote the position vector of a point with respect to the G coordinate system and let $\boldsymbol{r}^{\mathrm{LG}}$ denote the position vector of the same point with respect to the LG system. Then we have

$$\boldsymbol{r}^{\mathrm{G}} = \boldsymbol{o}^{\mathrm{G}}_{\mathrm{LG}} + R_{\mathrm{LG}\to\mathrm{G}} \boldsymbol{r}^{\mathrm{LG}}, \tag{B7}$$

and

$$\boldsymbol{r}^{\mathrm{LG}} = R_{\mathrm{G}\to\mathrm{LG}}(\boldsymbol{r}^{\mathrm{G}} - \boldsymbol{o}^{\mathrm{G}}_{\mathrm{LG}}), \tag{B8}$$

where the relationship between $R_{\mathrm{G}\to\mathrm{LG}}$ and $R_{\mathrm{LG}\to\mathrm{G}}$ has been noted in (B5).

It is important to bring out the fact that the elements of these rotation matrices are dependent upon $\boldsymbol{o}^{\mathrm{G}}_{\mathrm{LG}}$, whether the position of this point is specified as a latitude, longitude and height, or in Cartesian coordinates. It is noted that the rotation matrices

are expressed more cleanly in terms of (ϕ, λ, h), but these may be obtained in terms of $(x^{\mathrm{G}}, y^{\mathrm{G}}, z^{\mathrm{G}})$ through the transformation defined in Appendix A.

When transforming the velocities between the G and LG systems the time derivative of the matrices $R_{\mathrm{G}\to\mathrm{LG}}$ and $R_{\mathrm{LG}\to\mathrm{G}}$ are required. The time derivative of $R_{\mathrm{G}\to\mathrm{LG}}$ is

$$\begin{aligned}\dot{R}_{\mathrm{G}\,\to\,\mathrm{LG}} = \dot{\phi}&\begin{pmatrix} 0 & 0 & 0 \\ -\cos\phi\cos\lambda & -\cos\phi\sin\lambda & -\sin\phi \\ -\sin\phi\cos\lambda & -\sin\phi\sin\lambda & \cos\phi \end{pmatrix} \\ +\dot{\lambda}&\begin{pmatrix} -\cos\lambda & -\sin\lambda & 0 \\ \sin\phi\sin\lambda & -\sin\phi\cos\lambda & 0 \\ -\cos\phi\sin\lambda & \cos\phi\cos\lambda & 0 \end{pmatrix}.\end{aligned} \tag{B9}$$

The time derivative of $R_{\mathrm{LG}\to\mathrm{G}}$ is the transpose of this.

Appendix C - The rotation matrix in terms of the Euler angles

The rotation transformation from one frame to another is considered as composed of three rotations, denoted by $\Omega = (\Omega_x, \Omega_y, \Omega_z)$. The convention for defining these rotations and the expression for the rotation matrix are defined in this appendix.

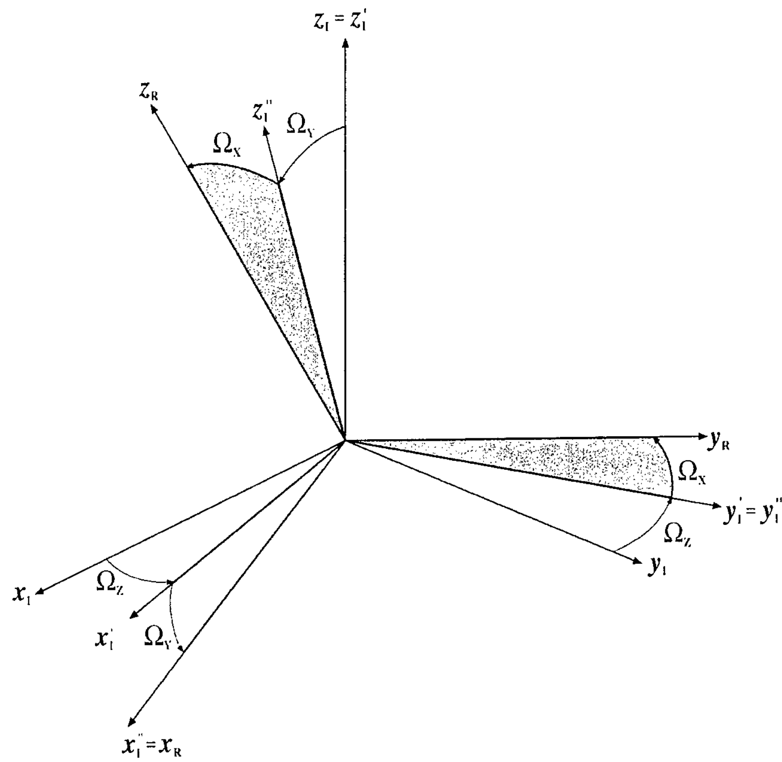

Figure C1: The Euler angles.

Let $\boldsymbol{r}^{\mathrm{I}}$ denote the components of a vector with respect to the coordinate system denoted by I, and suppose that after transformation through a rotation described by the Euler angle set Ω the components of this vector are $\boldsymbol{r}^{\mathrm{R}}$. The first rotation is Ω_z about the z_{I}-axis, measured positive as clockwise when looking along the axis and illustrated in Figure C1. All other rotations are defined using this convention. The matrix of this rotation is

$$\begin{pmatrix} \cos\Omega_z & \sin\Omega_z & 0 \\ -\sin\Omega_z & \cos\Omega_z & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

The second rotation is Ω_y about the y_{I}'-axis, as illustrated in Figure C1. Here the superscript $'$ indicates the coordinate system obtained after the first rotation. The matrix of this rotation is

$$\begin{pmatrix} \cos\Omega_y & 0 & -\sin\Omega_y \\ 0 & 1 & 0 \\ \sin\Omega_y & 0 & \cos\Omega_y \end{pmatrix}.$$

The final rotation is Ω_x about the y_{I}''-axis, as illustrated in Figure C1. The superscript $''$ indicates the coordinate system obtained after two rotations. The matrix of this final

rotation is

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\Omega_x & \sin\Omega_x \\ 0 & -\sin\Omega_x & \cos\Omega_x \end{pmatrix}.$$

Composing the matrices gives the transformation as

$$\boldsymbol{r}^{\mathrm{R}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\Omega_x & \sin\Omega_x \\ 0 & -\sin\Omega_x & \cos\Omega_x \end{pmatrix} \begin{pmatrix} \cos\Omega_y & 0 & -\sin\Omega_y \\ 0 & 1 & 0 \\ \sin\Omega_y & 0 & \cos\Omega_y \end{pmatrix} \begin{pmatrix} \cos\Omega_z & \sin\Omega_z & 0 \\ -\sin\Omega_z & \cos\Omega_z & 0 \\ 0 & 0 & 1 \end{pmatrix} \boldsymbol{r}^{\mathrm{I}}. \tag{C1}$$

Evaluating the matrix product gives the rotation matrix as

$$R = \begin{pmatrix} \cos\Omega_y\cos\Omega_z & \cos\Omega_y\sin\Omega_z & -\sin\Omega_y \\ -\cos\Omega_x\sin\Omega_z + \sin\Omega_x\sin\Omega_y\cos\Omega_z & \cos\Omega_x\cos\Omega_z + \sin\Omega_x\sin\Omega_y\sin\Omega_z & \sin\Omega_x\cos\Omega_y \\ \sin\Omega_x\sin\Omega_z + \cos\Omega_x\sin\Omega_y\cos\Omega_z & -\sin\Omega_x\cos\Omega_z + \cos\Omega_x\sin\Omega_y\sin\Omega_z & \cos\Omega_x\cos\Omega_y \end{pmatrix}. \tag{C2}$$

The time derivative of the rotation matrix is

$$\begin{aligned} \dot{R} = \dot{\Omega}_x &\begin{pmatrix} 0 & 0 & 0 \\ \sin\Omega_x\sin\Omega_z + \cos\Omega_x\sin\Omega_y\cos\Omega_z & -\sin\Omega_x\cos\Omega_z + \cos\Omega_x\sin\Omega_y\sin\Omega_z & \cos\Omega_x\cos\Omega_y \\ \cos\Omega_x\sin\Omega_z - \sin\Omega_x\sin\Omega_y\cos\Omega_z & -\cos\Omega_x\cos\Omega_z - \sin\Omega_x\sin\Omega_y\sin\Omega_z & -\sin\Omega_x\cos\Omega_y \end{pmatrix} \\ + \dot{\Omega}_y &\begin{pmatrix} -\sin\Omega_y\cos\Omega_z & -\sin\Omega_y\sin\Omega_z & -\cos\Omega_y \\ \sin\Omega_x\cos\Omega_y\cos\Omega_z & \sin\Omega_x\cos\Omega_y\sin\Omega_z & -\sin\Omega_x\sin\Omega_y \\ \cos\Omega_x\cos\Omega_y\cos\Omega_z & \cos\Omega_x\cos\Omega_y\sin\Omega_z & -\cos\Omega_x\sin\Omega_y \end{pmatrix} \\ + \dot{\Omega}_z &\begin{pmatrix} -\cos\Omega_y\sin\Omega_z & \cos\Omega_y\cos\Omega_z & 0 \\ -\cos\Omega_x\cos\Omega_z - \sin\Omega_x\sin\Omega_y\sin\Omega_z & -\cos\Omega_x\sin\Omega_z + \sin\Omega_x\sin\Omega_y\cos\Omega_z & 0 \\ \sin\Omega_x\cos\Omega_z - \cos\Omega_x\sin\Omega_y\sin\Omega_z & \sin\Omega_x\sin\Omega_z + \cos\Omega_x\sin\Omega_y\cos\Omega_z & 0 \end{pmatrix}. \end{aligned} \tag{C3}$$

Appendix D - Recovering the Euler angles from a rotation matrix

Suppose we have the rotation matrix R with elements

$$R = \begin{pmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{pmatrix}. \tag{D1}$$

Equating (D1) and (C2) we have

$$\sin \Omega_y = -R_{13}. \tag{D2}$$

We may choose $\Omega_y \in [-\pi/2, \pi/2]$ so

$$\cos \Omega_y = (1 - R_{13}^2)^{1/2}. \tag{D3}$$

Provided $|R_{13}| \neq 1$, that is $\Omega_y \neq \pm\pi/2$, we have

$$\cos \Omega_z = R_{11}/(1 - R_{13}^2)^{1/2}, \tag{D4}$$

$$\sin \Omega_z = R_{12}/(1 - R_{13}^2)^{1/2}, \tag{D5}$$

$$\sin \Omega_x = R_{23}/(1 - R_{13}^2)^{1/2}, \tag{D6}$$

$$\cos \Omega_x = R_{33}/(1 - R_{13}^2)^{1/2}. \tag{D7}$$

If we define the arccos function on the interval $[0, \pi]$ then the sign of the sin function indicates whether the angle is positive or negative. From (D1) and (C2) we deduce that $\mathrm{sgn}(\sin \Omega_z) = \mathrm{sgn}(R_{12})$ and $\mathrm{sgn}(\sin \Omega_x) = \mathrm{sgn}(R_{23})$. Thus

$$\Omega_x = \mathrm{sgn}(R_{23}) \arccos(R_{33}/(1 - R_{13}^2)^{1/2}), \tag{D8}$$

$$\Omega_z = \mathrm{sgn}(R_{12}) \arccos(R_{11}/(1 - R_{13}^2)^{1/2}). \tag{D9}$$

For completeness, consider the case where $|R_{13}| = 1$, so $\Omega_y = \pm\pi/2$. In the event that $\Omega_y = \pi/2$ the rotation matrix is

$$R = \begin{pmatrix} 0 & 0 & -1 \\ \sin(\Omega_x - \Omega_z) & \cos(\Omega_x - \Omega_z) & 0 \\ \cos(\Omega_x - \Omega_z) & -\sin(\Omega_x - \Omega_z) & 0 \end{pmatrix}. \tag{D10}$$

From this we deduce that

$$\Omega_x - \Omega_z = \mathrm{sgn}(R_{21}) \arccos R_{22}. \tag{D11}$$

Hence Ω_x and Ω_z can be chosen to be any values that satisfy (D11). This follows from the fact that for a coordinate system that is derived from another through a sequence of rotations that includes $\Omega_y = \pi/2$, there is no unique way of expressing the transformation in terms of the Euler angles. However, in practical circumstances there may be other criteria that determine the values chosen for Ω_x and Ω_z. Continuity of the Euler angles may be significant in this regard.

In the case $\Omega_y = -\pi/2$ the rotation matrix becomes

$$R = \begin{pmatrix} 0 & 0 & 1 \\ -\sin(\Omega_x + \Omega_z) & \cos(\Omega_x + \Omega_z) & 0 \\ -\cos(\Omega_x + \Omega_z) & -\sin(\Omega_x + \Omega_z) & 0 \end{pmatrix}, \tag{D12}$$

from which we deduce

$$\Omega_x + \Omega_z = -\mathrm{sgn}(R_{12}) \arccos R_{22}. \tag{D13}$$

Appendix E - Recovering the time derivative of the Euler angles from the time derivative of a rotation matrix

Taking the time derivative of (D1) and equating with (C3) yields

$$\dot{R}_{11} = -\dot{\Omega}_y \sin\Omega_y \cos\Omega_z - \dot{\Omega}_z \cos\Omega_y \sin\Omega_z, \tag{E1}$$

$$\dot{R}_{12} = -\dot{\Omega}_y \sin\Omega_y \sin\Omega_z + \dot{\Omega}_z \cos\Omega_y \cos\Omega_z, \tag{E2}$$

$$\dot{R}_{13} = -\dot{\Omega}_y \cos\Omega_y, \tag{E3}$$

$$\dot{R}_{23} = \dot{\Omega}_x \cos\Omega_x \cos\Omega_y - \dot{\Omega}_y \sin\Omega_x \sin\Omega_y, \tag{E4}$$

$$\dot{R}_{33} = -\dot{\Omega}_x \sin\Omega_x \cos\Omega_y - \dot{\Omega}_y \cos\Omega_x \sin\Omega_y. \tag{E5}$$

Assume that $\cos\Omega_y \neq 0$, that is $\Omega_y = \pm\pi/2$. From (E3)

$$\dot{\Omega}_y = -\dot{R}_{13}/\cos\Omega_y. \tag{E6}$$

Multiplying (E1) by $\sin\Omega_z$ and (E2) by $\cos\Omega_z$ and taking the difference yields

$$\dot{\Omega}_z = (\dot{R}_{12}\cos\Omega_z - \dot{R}_{11}\sin\Omega_z)/\cos\Omega_y. \tag{E7}$$

Similarly, multiplying (E4) by $\cos\Omega_x$, (E5) by $\sin\Omega_x$ and taking the difference yields

$$\dot{\Omega}_x = (\dot{R}_{23}\cos\Omega_x - \dot{R}_{33}\sin\Omega_x)/\cos\Omega_y. \tag{E8}$$

Assembling these results, provided $\cos\Omega_y \neq 0$,

$$\begin{aligned} \dot{\Omega}_x &= (\dot{R}_{23}\cos\Omega_x - \dot{R}_{33}\sin\Omega_x)/\cos\Omega_y, \\ \dot{\Omega}_y &= -\dot{R}_{13}/\cos\Omega_y, \\ \dot{\Omega}_z &= (\dot{R}_{12}\cos\Omega_z - \dot{R}_{11}\sin\Omega_z)/\cos\Omega_y. \end{aligned} \tag{E9}$$

For completeness, consider the case where $\cos\Omega_y = 0$, so $\Omega_y = \pm\pi/2$. (E1)-(E5) become

$$\begin{aligned} \dot{R}_{11} &= -\dot{\Omega}_y \sin\Omega_y \cos\Omega_z, \\ \dot{R}_{12} &= -\dot{\Omega}_y \sin\Omega_y \sin\Omega_z, \\ \dot{R}_{13} &= 0, \\ \dot{R}_{23} &= -\dot{\Omega}_y \sin\Omega_x \sin\Omega_y, \\ \dot{R}_{33} &= -\dot{\Omega}_y \cos\Omega_x \sin\Omega_y. \end{aligned} \tag{E10}$$

From these we can deduce the following equivalent expressions for $\dot{\Omega}_y$:

$$\dot{\Omega}_y = -(\dot{R}_{11}\cos\Omega_z + \dot{R}_{12}\sin\Omega_z)/\sin\Omega_y, \tag{E11}$$

$$\dot{\Omega}_y = -(\dot{R}_{23}\sin\Omega_x + \dot{R}_{33}\cos\Omega_x)/\sin\Omega_y, \tag{E12}$$

noting that $\sin\Omega_y = \pm 1$.

To obtain expressions for $\dot{\Omega}_x$ and $\dot{\Omega}_z$ we need to consider the expressions for the time derivatives of the remaining elements of the rotation matrix. We have in the case $\Omega_y = \pi/2$

$$\begin{aligned}
\dot{R}_{21} &= (\dot{\Omega}_x - \dot{\Omega}_z)\cos(\Omega_x - \Omega_z),\\
\dot{R}_{22} &= -(\dot{\Omega}_x - \dot{\Omega}_z)\sin(\Omega_x - \Omega_z),\\
\dot{R}_{31} &= -(\dot{\Omega}_x - \dot{\Omega}_z)\sin(\Omega_x - \Omega_z),\\
\dot{R}_{32} &= -(\dot{\Omega}_x - \dot{\Omega}_z)\cos(\Omega_x - \Omega_z).
\end{aligned} \tag{E13}$$

In the case $\Omega_y = -\pi/2$

$$\begin{aligned}
\dot{R}_{21} &= -(\dot{\Omega}_x + \dot{\Omega}_z)\cos(\Omega_x + \Omega_z),\\
\dot{R}_{22} &= -(\dot{\Omega}_x + \dot{\Omega}_z)\sin(\Omega_x + \Omega_z),\\
\dot{R}_{31} &= (\dot{\Omega}_x + \dot{\Omega}_z)\sin(\Omega_x + \Omega_z),\\
\dot{R}_{32} &= -(\dot{\Omega}_x + \dot{\Omega}_z)\cos(\Omega_x + \Omega_z).
\end{aligned} \tag{E14}$$

Note that we can substitute for $\sin(\Omega_x - \Omega_z)$ and $\cos(\Omega_x - \Omega_z)$ using (D10) and for $\sin(\Omega_x + \Omega_z)$ and $\cos(\Omega_x + \Omega_z)$ using (D12). Hence we get for the case $\Omega_y = \pi/2$

$$\begin{aligned}
\dot{R}_{21} &= R_{22}(\dot{\Omega}_x - \dot{\Omega}_z),\\
\dot{R}_{22} &= -R_{21}(\dot{\Omega}_x - \dot{\Omega}_z),\\
\dot{R}_{31} &= R_{32}(\dot{\Omega}_x - \dot{\Omega}_z),\\
\dot{R}_{32} &= -R_{31}(\dot{\Omega}_x - \dot{\Omega}_z),
\end{aligned} \tag{E15}$$

and in the case $\Omega_y = -\pi/2$

$$\begin{aligned}
\dot{R}_{21} &= -R_{22}(\dot{\Omega}_x + \dot{\Omega}_z),\\
\dot{R}_{22} &= R_{21}(\dot{\Omega}_x + \dot{\Omega}_z),\\
\dot{R}_{31} &= -R_{32}(\dot{\Omega}_x + \dot{\Omega}_z),\\
\dot{R}_{32} &= R_{31}(\dot{\Omega}_x + \dot{\Omega}_z).
\end{aligned} \tag{E16}$$

Given that R is a rotation matrix, not all the coefficients on the right hand side of (E15) and (E16) can be zero so a unique solution can be obtained for $\dot{\Omega}_x - \dot{\Omega}_z$ in the case where $\Omega_y = \pi/2$ and for $\dot{\Omega}_x + \dot{\Omega}_z$ in the case where $\Omega_y = -\pi/2$. As for the case of determining the Euler angles given the rotation matrix, the freedom in solution for Ω_x and Ω_z may be exploited and perhaps used to ensure continuity of $\dot{\Omega}_x$ and $\dot{\Omega}_z$.

Appendix F - The relationship between the time derivative of the Euler angles and the angular velocity

Consider two Cartesian coordinate systems. The first is stationary and the second is in rotation with respect to it. At any instant, the transformation from the first frame to the second is described by a rotation matrix defined in terms of the Euler angles in the convention used previously. The Euler angles are functions of time. The rotation of the second coordinate system may be equivalently described by the angular velocity of this frame relative to the first. Let $\boldsymbol{v}^{\mathrm{I}}$ denote the velocity of a point with respect to the first frame. Let $\boldsymbol{v}^{\mathrm{R}}$ denote the velocity of this same point with respect to a second coordinate system that rotates with the angular velocity of the second coordinate system define above, but which is instantaneously cooincident with the first frame. We have

$$\boldsymbol{v}^{\mathrm{I}} = \boldsymbol{v}^{\mathrm{R}} + \boldsymbol{\omega} \times \boldsymbol{r}^{\mathrm{I}}, \tag{F1}$$

where $\boldsymbol{\omega}$ is the angular velocity of the rotating coordinate system measured with respect to the first coordinate system. $\boldsymbol{r}^{\mathrm{I}}$ is the position vector of the point given with respect to the first coordinate system, which is instantaneously the same as it would be given with respect to the rotating system.

Let $\boldsymbol{r}^{\mathrm{R}}$ be the position vector of the same point given with respect to the second coordinate system. We have

$$\boldsymbol{r}^{\mathrm{R}} = R_{\mathrm{I}\rightarrow\mathrm{R}}\boldsymbol{r}^{\mathrm{I}}. \tag{F2}$$

where $R_{\mathrm{I}\rightarrow\mathrm{R}}$ is the rotation matrix that associates the coordinate systems. Differentiating with respect to time we have

$$\dot{\boldsymbol{r}}^{\mathrm{R}} = \dot{R}_{\mathrm{I}\rightarrow\mathrm{R}}\boldsymbol{r}^{\mathrm{I}} + R_{\mathrm{I}\rightarrow\mathrm{R}}\dot{\boldsymbol{r}}^{\mathrm{I}}. \tag{F3}$$

In the second frame of reference the point is stationary so $\dot{\boldsymbol{r}}^{\mathrm{R}} = 0$ and

$$\dot{\boldsymbol{r}}^{\mathrm{I}} = -R^{-1}_{\mathrm{I}\rightarrow\mathrm{R}}\dot{R}_{\mathrm{I}\rightarrow\mathrm{R}}\boldsymbol{r}^{\mathrm{I}} = \omega \times \boldsymbol{r}^{\mathrm{I}}. \tag{F4}$$

Now

$$\omega \times \boldsymbol{r}^{\mathrm{I}} = (\omega_y z^{\mathrm{I}} - \omega_z y^{\mathrm{I}})\,\boldsymbol{i} + (\omega_z x^{\mathrm{I}} - \omega_x z^{\mathrm{I}})\,\boldsymbol{j} + (\omega_x y^{\mathrm{I}} - \omega_y x^{\mathrm{I}})\,\boldsymbol{k}, \tag{F5}$$

which can be written in matrix form as

$$\begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix} \boldsymbol{r}^{\mathrm{I}}. \tag{F6}$$

Using this is (F4) gives

$$R^{-1}_{\mathrm{I}\rightarrow\mathrm{R}}\dot{R}_{\mathrm{I}\rightarrow\mathrm{R}} = \begin{pmatrix} 0 & \omega_z & -\omega_y \\ -\omega_z & 0 & \omega_x \\ \omega_y & -\omega_x & 0 \end{pmatrix}. \tag{F7}$$

Carrying out the differentiation we get

$$\begin{aligned} \begin{pmatrix} 0 & \omega_z & -\omega_y \\ -\omega_z & 0 & \omega_x \\ \omega_y & -\omega_x & 0 \end{pmatrix} &= \dot{\Omega}_z \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\ &\quad + \dot{\Omega}_y \begin{pmatrix} 0 & 0 & -\cos\Omega_z \\ 0 & 0 & -\sin\Omega_z \\ \cos\Omega_z & \sin\Omega_z & 0 \end{pmatrix} \\ &\quad + \dot{\Omega}_x \begin{pmatrix} 0 & -\sin\Omega_y & -\cos\Omega_y\sin\Omega_z \\ \sin\Omega_y & 0 & \cos\Omega_y\sin\Omega_z \\ \cos\Omega_y\sin\Omega_z & -\cos\Omega_y\cos\Omega_z & 0 \end{pmatrix}. \end{aligned} \tag{F8}$$

Hence

$$\begin{aligned}\omega_x &= \dot{\Omega}_x \cos\Omega_y \cos\Omega_z - \dot{\Omega}_y \sin\Omega_z,\\ \omega_y &= \dot{\Omega}_x \cos\Omega_y \sin\Omega_z + \dot{\Omega}_y \cos\Omega_z,\\ \omega_z &= \dot{\Omega}_z - \dot{\Omega}_x \sin\Omega_y.\end{aligned} \tag{F9}$$

This recovers the formulae given in [4] (page 610).

It is often required to have the components of the angular velocity with respect the the rotating coordinate system. Using the rotation matrix $R_{\mathrm{I}\to\mathrm{R}}$ to effect the transformation gives

$$\begin{aligned}\omega_x^{\mathrm{R}} &= \dot{\Omega}_x - \dot{\Omega}_z \sin\Omega_y,\\ \omega_y^{\mathrm{R}} &= \dot{\Omega}_z \sin\Omega_x \cos\Omega_y + \dot{\Omega}_y \cos\Omega_x,\\ \omega_z^{\mathrm{R}} &= \dot{\Omega}_z \cos\Omega_x \cos\Omega_y - \dot{\Omega}_y \sin\Omega_x,\end{aligned} \tag{F10}$$

where the superscript $^{\mathrm{R}}$ indicates that the components are given with respect to the second coordinate system. This recovers the formulae given in [4] (page 609).

For completeness it is necessary to comment on the case where $\Omega_y = \pm\pi/2$ as in this case there is, given a rotation matrix and its time derivative, indeterminancy in Ω_x, Ω_z, $\dot{\Omega}_x$ and $\dot{\Omega}_z$. If $\Omega_y = \pi/2$ then the angular velocity is

$$\boldsymbol{\omega} = \begin{pmatrix} -\dot{\Omega}_y \sin\Omega_z \\ \dot{\Omega}_y \cos\Omega_z \\ \dot{\Omega}_z - \dot{\Omega}_x \end{pmatrix}, \tag{F11}$$

and if $\Omega_y = -\pi/2$ the angular velocity is

$$\boldsymbol{\omega} = \begin{pmatrix} -\dot{\Omega}_y \sin\Omega_z \\ \dot{\Omega}_y \cos\Omega_z \\ \dot{\Omega}_z + \dot{\Omega}_x \end{pmatrix}. \tag{F12}$$

Using (E10) and (E15) we get for the case $\Omega_y = \pi/2$

$$\boldsymbol{\omega} = \begin{pmatrix} \dot{R}_{12} \\ -\dot{R}_{11} \\ \dot{R}_{22}/R_{21} \end{pmatrix}, \tag{F13}$$

provided $R_{21} \neq 0$. If $R_{21} = 0$ the last component of $\boldsymbol{\omega}$ is replaced by $-\dot{R}_{21}/R_{22}$ which is guaranteed non-zero in this case.

Similarly, using (E10) and (E16), we get for the case $\Omega_y = -\pi/2$

$$\boldsymbol{\omega} = \begin{pmatrix} -\dot{R}_{12} \\ \dot{R}_{11} \\ \dot{R}_{22}/R_{21} \end{pmatrix}, \tag{F14}$$

provided $R_{21} \neq 0$. If $R_{21} = 0$ the last component of $\boldsymbol{\omega}$ is replaced by $-\dot{R}_{21}/R_{22}$ which is guaranteed non-zero in this case.

Hence we see that although there is indeterminancy in two of the Euler angles and their time derivatives in this case, the components of the angular velocity are unique, as they should be given they reflect a rotational motion assumed continuous.

DISTRIBUTION LIST

Coordinate Usage in the Virtual Ship Architecture - Issue 1.00

John P. Best

AUSTRALIA

DEFENCE ORGANISATION

S&T Program

Chief Defence Scientist
FAS Science Policy
AS Science Corporate Management
(shared copy)
Director General Science Policy Development
Counsellor Defence Science, London (Doc Data Sheet)
Counsellor Defence Science, Washington (Doc Data Sheet)
Scientific Adviser to MRDC Thailand (Doc Data Sheet)
Scientific Adviser Policy and Command
Navy Scientific Adviser
Scientific Adviser - Army (Doc Data Sheet and distribution list)
Air Force Scientific Adviser
Director Trials

Aeronautical and Maritime Research Laboratory

Director
CMOD
Dr J. Riley (MOD)
Dr J. Best (MOD)
A. Cramp (MOD)
N. Luckman (WSD)
Dr D. Sutton (WSD)
M. Saunders (WSD)
Dr A. Anvar (MOD)

Electronics and Surveillance Research Laboratory

Director
Dr J. Legg (SSD)
Dr S. Canney (EWD)
B. Hermans (EWD)
G. Horsfall (EWD)

DSTO Library and Archives

Library Fishermans Bend (Doc Data Sheet)
Library Maribyrnong (Doc Data Sheet)
Library Salisbury
Australian Archives
Library, MOD, Pyrmont
Library, MOD, HMAS Stirling
US Defense Technical Information Center, 2 copies
UK Defence Research Information Centre, 2 copies

Canada Defence Scientific Information Service
NZ Defence Information Centre
National Library of Australia

Capability Systems Staff
Director General Maritime Development
Director General C3I Development (Doc Data Sheet)
Director General Aerospace Development (Doc Data Sheet)

Navy
Capability Development Manager, SCFEG, Building 90, Garden Island
SO (Science), COMAUSNAVSURFGRP, Garden Island (Doc Data Sheet)

Army
ASNSO ABCA, Puckapunyal, (4 copies)
SO (Science), DJFHQ(L), MILPO Enoggera, Queensland 4051 (Doc Data Sheet)

Intelligence Program
DGSTA Defence Intelligence Organisation
Manager, Information Centre, Defence Intelligence Organisation

Corporate Support Program
Library Manager, DLS-Canberra

UNIVERSITIES AND COLLEGES
Australian Defence Force Academy
Library
Head of Aerospace and Mechanical Engineering
Serials Section (M list), Deakin University Library, Geelong, 3217
Hargrave Library, Monash University (Doc Data Sheet)
Librarian, Flinders University

OTHER ORGANISATIONS
NASA (Canberra)
AusInfo

OUTSIDE AUSTRALIA

ABSTRACTING AND INFORMATION ORGANISATIONS
Library, Chemical Abstracts Reference Service
Engineering Societies Library, US
Materials Information, Cambridge Scientific Abstracts, US
Documents Librarian, The Center for Research Libraries, US

INFORMATION EXCHANGE AGREEMENT PARTNERS
Acquisitions Unit, Science Reference and Information Service, UK
Library - Exchange Desk, National Institute of Standards and Technology, US

SPARES (5 copies)

Total number of copies: 57

Page classification: UNCLASSIFIED

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION **DOCUMENT CONTROL DATA**	1. PRIVACY MARKING/CAVEAT (OF DOCUMENT)

2. TITLE	3. SECURITY CLASSIFICATION (FOR UNCLASSIFIED REPORTS THAT ARE LIMITED RELEASE USE (L) NEXT TO DOCUMENT CLASSIFICATION)
Coordinate Usage in the Virtual Ship Architecture Issue 1.00	Document (U) Title (U) Abstract (U)
4. AUTHOR(S)	5. CORPORATE AUTHOR
John P. Best	Aeronautical and Maritime Research Laboratory PO Box 4331 Melbourne Vic 3001 Australia

6a. DSTO NUMBER	6b. AR NUMBER	6c. TYPE OF REPORT	7. DOCUMENT DATE
DSTO-GD-0260	AR-011-626	General Document	October 2000

8. FILE NUMBER	9. TASK NUMBER	10. TASK SPONSOR	11. NO. OF PAGES	12. NO. OF REFERENCES
M9505-19-166	NAV 98/173	DGMD	30	4

13. URL ON THE WORLD WIDE WEB	14. RELEASE AUTHORITY
http://www.dsto.defence.gov.au/corporate/reports/DSTO-GD-0260.pdf	Chief, Maritime Operations Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for public release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, SALISBURY, SA 5108

16. DELIBERATE ANNOUNCEMENT

No Limitations

17. CASUAL ANNOUNCEMENT Yes

18. DEFTEST DESCRIPTORS

Command and control systems, Computerized simulation, Computer architecture, Naval warfare, Virtual reality

19. ABSTRACT

This document describes the coordinate systems used within the Virtual Ship Architecture (VSA) and defines object attributes that convey kinematic information.

Page classification: UNCLASSIFIED